An Improved Estimation Methodology for Hybrid BIST Cost Calculation

Gert Jervan, Zebo Peng
Linköping University
Embedded Systems Laboratory
{gerje, zpe}@ida.liu.se

Raimund Ubar, Olga Korelina
Tallinn University of Technology
Department of Computer Engineering
raiub@pld.ttu.ee

Abstract[1]:

This paper presents an improved estimation methodology for hybrid BIST cost calculation. In a hybrid BIST approach the test set is assembled from pseudorandom and deterministic test patterns. The efficiency of the hybrid BIST approach is largely determined by the ratio of those test patterns in the final test set. Unfortunately exact algorithms for finding the test sets are computationally very expensive. Therefore in this paper we propose an improved estimation methodology for fast calculation of the hybrid test set. The methodology is based on real fault simulation results and experimental results have shown that the method is more accurate than the statistical method proposed earlier.

1. Introduction

Testing of systems-on-chip (SoC) is a problematic and time consuming task, mainly due to their complexity and high integration density [1]. To test the individual cores of a SoC the test pattern source and sink have to be available together with an appropriate test access mechanism (TAM) [2]. Due to the rapid increase of chip speed and test data volume, the traditional Automatic Test Equipment (ATE) based solution is becoming increasingly expensive and inaccurate. Therefore, in order to apply at-speed tests and to keep the test costs under control, built-in self-test (BIST) solutions are becoming a mainstream technology for testing such complex systems.

BIST for digital logic (logic BIST) uses mostly pseudorandom tests. Due to several reasons, like very long test sequences, and random pattern resistant faults, this approach may not always be efficient. One solution to the problem is to complement pseudorandom test patterns with deterministic test patterns, applied from the on-chip memory or, in special situations, from the ATE. This approach is usually referred to as hybrid BIST [3].

One of the important parameters influencing the efficiency of a hybrid BIST approach is the ratio of pseudorandom and deterministic test patterns in the final test set. As the amount of resources on the chip is limited, the final test set has to be designed in such a way that the deterministic patterns fit into the on-chip memory. At the same time the testing time must be minimized in order to reduce testing cost and time-to-market.

Finding the best ratio of those test sets is computationally very expensive and therefore we have developed a methodology for fast estimation of the ratio of the pseudorandom and deterministic test patterns in the final hybrid test set.

The rest of the paper is organized as follows. In the following section an overview of the hybrid BIST is given. It is followed by the hybrid BIST cost calculation explanation in Section 3. Section 4 describes the estimation methodology that is illustrated with experimental results in Section 5. The paper is concluded in Section 6.

2. Hybrid BIST

In general a hybrid BIST approach combines two different types of tests. It starts with a pseudorandom test sequence of length L and continues with precomputed deterministic test patterns, stored in the system, in order to reach the desirable fault coverage. For off-line generation of the deterministic test patterns, arbitrary software test generators may be used, based on deterministic, random or genetic algorithms.

The length L of the pseudorandom test is an important parameter, which determines the behaviour of the whole test process. A shorter pseudorandom test set implies a larger deterministic test set. This however requires additional memory space, but at the same time, shortens the overall testing time. A longer pseudorandom test, on the other hand, will lead to longer test application time with reduced memory requirements. Therefore it is crucial to determine the optimal length of pseudorandom test in order to minimize the total testing cost.

Figure 1 illustrates graphically calculation of the total cost of the hybrid BIST consisting of pseudorandom test patterns and stored test patterns, generated off-line. We can define the total test cost of the hybrid BIST C_{TOTAL} as:

$$C_{TOTAL} = C_{GEN} + C_{MEM} = \alpha L + \beta S$$

[1] This work has been supported by the EC project EVIKINGS (IST-2001-37592), Estonian Science Foundation Grants 4300 and 5649, and by the Swedish Foundation for Strategic Research (SSF) under the Strategic Integrated Electronic Systems Research (STRINGENT) program.

where C_{GEN} is the cost related to the time for generating L pseudorandom test patterns (number of clock cycles), C_{MEM} is the memory cost for storing S precomputed test patterns (number of stored test patterns), and a, b are constants to map the test length and memory space to the costs of the two parts of the test solutions to be mixed.

Figure 1 describes how the cost C_{GEN} of pseudorandom test is increasing when striving to higher fault coverage.

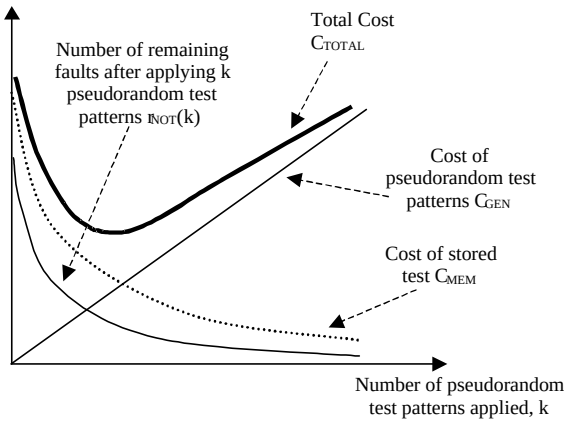

Fig. 1. Cost calculation for hybrid BIST

In general, it can be very expensive to achieve high fault coverage with pseudorandom test patterns only. The curve C_{MEM} describes the cost that we have to pay for storing precomputed tests at the given fault coverage reached by pseudorandom testing. The total cost C_{TOTAL} is the sum of the two mentioned costs aL and bS. The weights a and b reflect the correlation between the cost and the pseudorandom test time (number of clock cycles used) or between the cost and the memory size needed for storing the precomputed test sequence.

3. Cost Calculation for Hybrid BIST

The main purpose of this work is to propose a fast method for calculating the number of additional deterministic test patterns S for any arbitrary number of pseudorandom test patterns in order to reach maximum obtainable fault coverage.

Creating the curve C_{GEN} is not difficult. For this purpose, a simulation of the behaviour of the LSFR, used for pseudorandom test pattern generation, is needed. A fault simulation should be carried out for the complete test sequence generated by the LFSR. As a result of such a simulation, we find for each clock cycle the list of faults which were covered up to this clock cycle. By removing these faults from the complete fault list, we will know the number of faults remaining to be tested.

More difficult is to find the values of bS, the cost for storing additional deterministic patterns in order to reach the given fault coverage level (100% in the ideal case). In [3] we proposed a method based on repetitive use of the ATPG and in [4] a method based on fault table manipulations was described. Both procedures are accurate but time-consuming and therefore not feasible for larger designs.

To overcome the complexity explosion problem we have developed an estimation methodology, that leads us to the approximate solution. This can be used as an initial solution for a search of more accurate results, using different optimization heuristics. In [5] a method based on Tabu search has been proposed.

The previously proposed estimation method was based on statistical information and was therefore not always very accurate. In this paper we will propose a new, improved method, that is based on fault simulation results of the particular design and, as we will demonstrate with experimental results, has proven to be more accurate.

4. Test Cost Estimation Methodology

Let us denote the deterministic test set with *TD* and efficient pseudorandom test set [6] with *TPE*. In the following we will use *FD(i)* and *FPE(i)* to denote the fault coverage figures of the test sequences *TD*(*i*) and *TPE*(*i*), correspondingly, where *i* is the length of the test sequence.

Procedure 1: Estimation of the length of the deterministic test set TD.

1. Calculate, by fault simulation, the fault coverage functions $FD(i)$, $i = 1, 2, \ldots, |TD|$, and $FPE(i)$, $i = 1, 2, \ldots, |TPE|$. The patterns in *TD* are ordered in such the way that each pattern put into the sequence contributes with maximum increase in fault coverage.
2. For each $i^* \le |TPE|$, find the fault coverage value F^* that can be reached by a sequence of patterns $(P_1, P_2, \ldots, P_{i^*}) \subseteq TPE$ (see Figure 2).
3. By solving the equation $FD(i) = F^*$, find the maximum integer value j^* that satisfies the condition $FD(j^*) \le F^*$. The value of j^* is the length of the deterministic sequence that can achieve the same fault coverage F^*.
4. Calculate the value of $|TD^E(i^*)| = |TD| - j^*$ which is the number of test patterns needed from the *TD* to reach to the maximum achievable fault coverage.

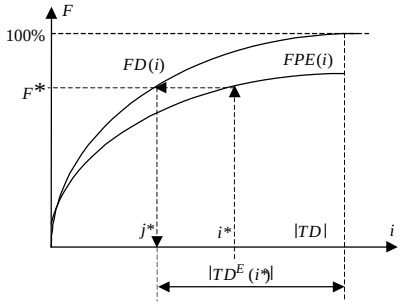

Fig. 2. Estimation of the length of the deterministic test sequence

The value $|TD^E(i^*)|=|TD|- j^*$, calculated by the Procedure 1, can be used to estimate the length of the deterministic test sequence TD^* in the hybrid test set $TH = \{TP^*, TD^*\}$ with i^* efficient test patterns in TP^*.

By finding $|TD^E(j)|$ for all $j = 1, 2, \ldots, |TPE|$ we get the cost function estimate $C^E_{MEM}(j)$.

In the following we will illustrate the procedure 1 with an example. In Figure 3 we have presented an extract of the

fault simulation results for both test sets. The length of the pseudorandom sequence has to be only so long as potentially necessary. By knowing the length of the complete deterministic test set and fault coverage figures for every individual pattern we can estimate the size of the additional deterministic test set for any length of the pseudorandom test sequence, as illustrated in the Figure 3. Here we can see that for a given core 60 deterministic test cycles are needed to obtain the same fault coverage as 524 pseudorandom test cycles and it requires additional 30 deterministic test cycles to reach 100% fault coverage. Based on this information we assume, that if we will apply those 30 deterministic test cycles on top of the 524 pseudorandom cycles, we can obtain close to the maximum fault coverage.

|TP|	*FC%*	*|TD|*	*FC%*
1	21.9	1	43.3
2	34.7	2	45.6
	…		
524	97.5	60	97.5
	…		
1000	98.9	90	100

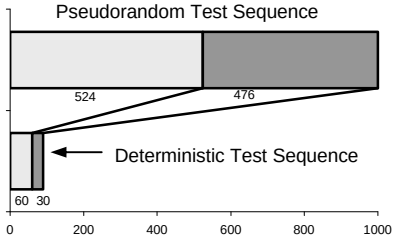

Fig. 3. Estimation of the length of the deterministic test sequence

5. Experimental Results

We have performed experiments with all designs from the ISCAS85 benchmark family. Some of those results are illustrated in Figure 4. In those charts we have depicted the memory requirement (the size of the deterministic test set) for every pseudorandom test length. Obviously – the longer the pseudorandom test sequence is, the smaller is the memory requirement. We have compared our earlier estimation methodology [3] against the estimation methodology proposed in this paper. In addition we have also depicted the real memory cost. This has been obtained by the repetitive use of the ATPG [3]. As it can be seen from the results, a new estimation methodology gives better estimates than the previous one (the curve "New Approximate" is much closer to the "Real" than the "Old Approximate"), mainly in the situations, when the hybrid test set contains smaller amount of pseudorandom test patterns.

This approximation methodology can be used in different contexts. It can be used for total test cot minimization for single-core designs (as described in [3], [4], [5]). We have also demonstrated, that the same estimation methodology can be used efficiently for test time minimization for multi-core designs, under tester memory constraints. The method has been proved to be efficient with combinatorial cores [6] as well as with sequential cores [7].

6. Conclusions

In this paper we have proposed an improved estimation methodology that can be used for hybrid BIST cost calculations. The exact calculations for finding the hybrid test set configuration is computationally expensive and therefore a fast estimation procedure might be highly useful.

As it was shown by experimental results, the improved estimation algorithm has produced significantly better results than the estimation method used earlier. It can be used as a good starting point for the search for a global optimum by a few additional exact calculations of the real cost.

References

[1] B. T Murray, J. P. Hayes, "Testing ICs: Getting to the core of the problem," *IEEE Transactions on Computer*, Vol. 29, pp. 32-39, November 1996.

[2] Y. Zorian, E. J. Marinissen, S. Dey, "Testing Embedded Core-Based System Chips," *IEEE International Test Conference (ITC'98)*, pp. 130-143, 1998.

[3] G. Jervan, Z. Peng, R. Ubar, "Test Cost Minimization for Hybrid BIST," *IEEE Int. Symp. on Defect and Fault Tolerance in VLSI Systems (DFT'00)*, pp.283-291, 2000.

[4] R. Ubar, G. Jervan, Z. Peng, E. Orasson, R. Raidma, "Fast Test Cost Calculation for Hybrid BIST in Digital Systems", *Euromicro Symposium on Digital Systems Design*, pp. 318-325, 2001.

[5] R. Ubar, H. Kruus, G. Jervan, Z. Peng, "Using Tabu Search Method for Optimizing the Cost of Hybrid BIST", *16th Conference on Design of Circuits and Integrated Systems (DCIS 2001)*, pp. 445-450´, 2001.

[6] G. Jervan, P. Eles, Z. Peng, R. Ubar, M. Jenihhin, "Test Time Minimization for Hybrid BIST of Core-Based Systems", *12th IEEE Asian Test Symposium (ATS03)*, pp. 318-323, 2003.

[7] G. Jervan, P. Eles, Z. Peng, R. Ubar, M. Jenihhin, "Hybrid BIST Time Minimization for Core-Based Systems with STUMPS Architecture", *18th IEEE International Symposium on Defect and Fault Tolerance in VLSI Systems (DFT'03)*, pp. 225-232, 2003.

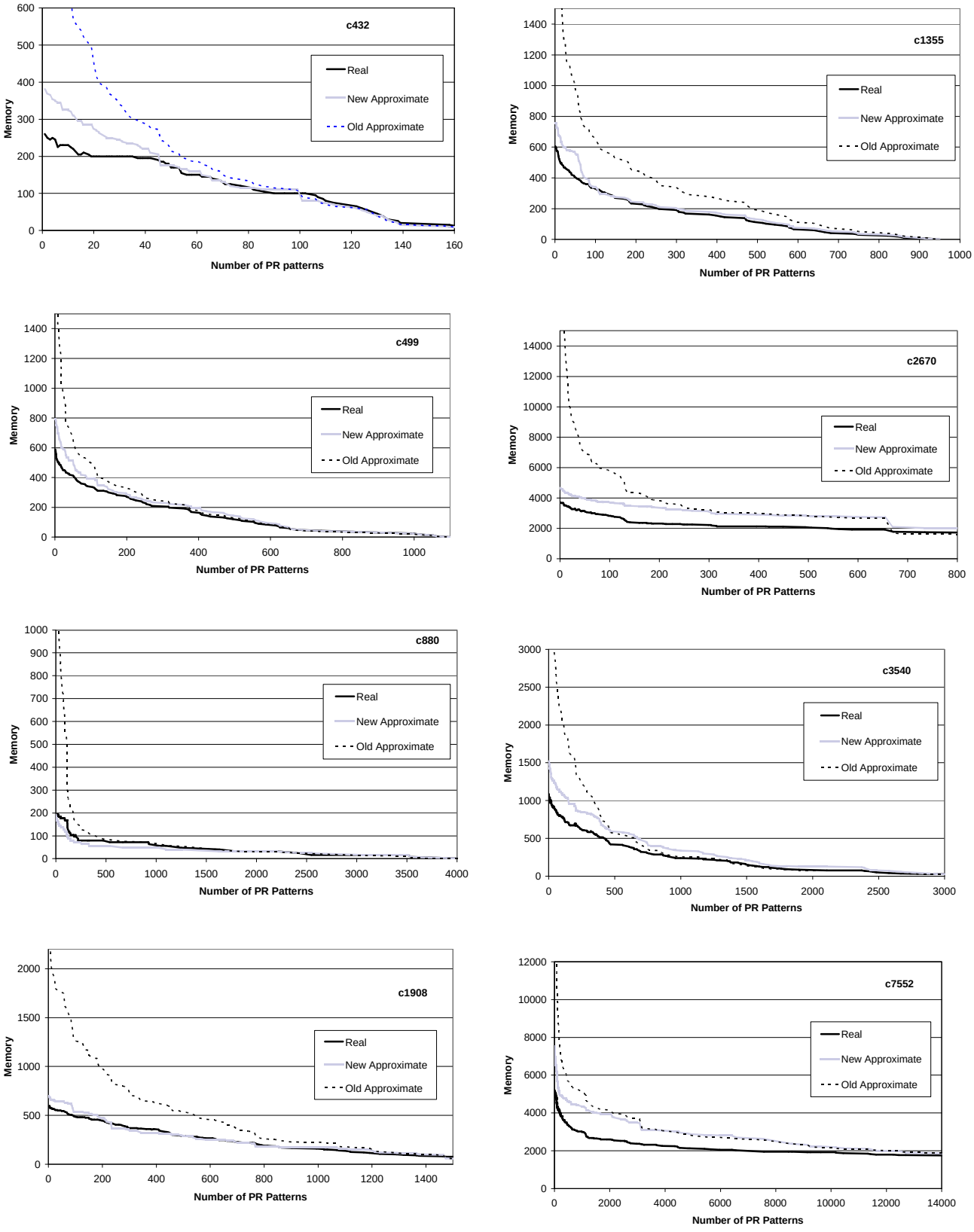

Fig. 4. Experimental results with ISCAS 85 benchmark designs